IN THE SPECIFICATION:

Please replace paragraph [0006] with the following amended paragraph:

[0006] In a first embodiment, a method of testing content is provided. The method includes parsing, by a parser, of two or more documents in tandem on an element-by-element basis, whereby the elements of each of the documents are sequentially parsed. Upon parsing an element in a first document of the two or more documents and a respective element in each of the other documents, the respective parsed elements are compared to one another. On the basis of the comparison, it is determined whether the documents are at least equivalent. In one embodiment, each of the other documents is a current response from an application responding to a submitted request and the first document is a control document retrieved from storage and previously returned from the application in response to the request.

Please replace paragraph [0032] with the following amended paragraph:

[0032] The SAMPLE ACTION/OUTPUT SEQUENCE above shows an illustrative sequence of user actions and corresponding output (in the form of XHTML) returned from the application 108. Illustratively, a user navigates to a Home page (Action 1→Page 1) and then ~~logins~~ logs in with the appropriate login ID and password and is presented with, for example, a query input page (Action 2→Page 2). The user then inputs and executes a query against the database 114 and is provided with any query results (Action 3→Page 3).

Please replace paragraph [0043] with the following amended paragraph:

[0043] FIG. 3 describes, in one aspect, comparing a control document to a live document. However, as will be described in more detail below, some embodiments (specifically, some embodiments of the test expression validation) do not involve a comparison of documents. Further, where a comparison is performed, more than two documents may be compared. That is, for a given control document and request, two or more live documents may be returned and compared to the ~~live~~ control document.

Please replace paragraph [0044] with the following amended paragraph:

[0044] Referring now to FIGURE 4, one embodiment of an element-by-element testing operation 310 is shown. In general, “element-by-element testing” refers to comparative testing between sequentially determined elements of at least two documents (i.e., a control document and a live document). By traversing and comparing documents in this manner a degree of structural equivalence between the documents can be determined. For example, the absence or presence of a given structure, such as a table, a button, or a border in each of the documents can be determined. Accordingly, structural equivalence refers to a correspondence in the layout of documents. In addition to determining a degree of structural equivalence between documents, content equivalence can be determined. That is, the absence or presence of specific content (e.g., table data) in the respective documents can also be determined. As noted above, element-by-element testing may be implemented using a SAX parser 122 and an appropriate comparator 124 (both shown in FIGURE 1). The structural testing operation begins (at step 402) by initiating parsing of the appropriate control document and response (i.e., the live document). Parsing the documents from beginning to end, the next sequential token is retrieved for each document (steps 404 and 406). In this context, a "token" is any document element of appropriate granularity to perform element-by-element testing. For example, where the documents are XHTML documents a token may be synonymous with a node (i.e., a tag) of the documents. For example, in the "OUTPUT DOCUMENT FOR A FIRST VERSION OF A DATABASE" shown above, the first node is “<html>”. For the two tokens from the respective documents one or more testing and validation techniques/modes (involving comparison of the tokens by the comparator 124) may be applied. In the embodiment illustrated by FIGURE 4, three different techniques are contemplated. Which of the three techniques is applied may be dependent upon the specific configuration settings of the UI testing tool 112. After the selected technique(s) is performed, the operation 310 determines whether the control document or the live document contains ~~anymore~~ any more tokens (step 414). If not, the operation ends; otherwise, processing continues with the next tokens from the control document and the live document (steps 404 and 406).

Please replace paragraph [0047] with the following amended paragraph:

[0047] Referring now to FIGURE 7 a third technique (step 412) is shown in which the UI testing tool 112 operates in an internationalization mode for the comparison of documents that should be structurally ~~the~~ equivalent, but are in different languages. In the illustrative embodiment, internationalization is accomplished by first determining whether the tokens of the control document and live document are character data. If so, the character data in both structures is consumed (step 704) by the parser. The comparator 124 then determines (step 706) whether the consumed character data is the same (or sufficiently similar to a predefined tolerance) in both documents. If not, the documents are assumed to be appropriately translated in their respective languages, and processing returns to step 414 FIGURE 4. On the other hand, if the character data is the same a warning is issued (708) about it possible mistranslation. Processing then returns to step 414 FIGURE 4.

Please replace paragraph [0048] with the following amended paragraph:

[0048] Returning to step 702, if the control document and the live document do not both contain character data, it is determined whether only one contains character data. If so, a problem is reported (712) since the "type" (i.e., character data type) of tokens being compared should be the same, although the languages are different. Processing then returns to step 414 FIGURE 4. If neither token contains character data, a simple comparison of the tokens is performed as was described above with respect to FIGURE 5.

Please replace paragraph [0049] with the following amended paragraph:

[0049] Referring now to FIGURE 8, one embodiment of a test expression validation method (step 312 of FIGURE 3) is shown. Upon initiating the method 312, the response (i.e., live document) received at step 308 of FIGURE 3 is parsed (step 802). Any appropriate, previously defined control variables 130 are then applied. The parser 122 then parses the control document (step 808). Then, the appropriate, previously defined test expressions are retrieved (step 810). The test expressions retrieved are those corresponding to the parsed control document. For each test expression (step 812), the expression is applied (step 814) and then a determination is made as to whether the expression is satisfied (step 816). As noted above, determining whether a particular test expression is satisfied may vary according to different embodiments. In one embodiment, a given test expression may be applied to both documents, after which the documents are compared based on the control ~~variables. to~~ variables to the documents. This approach may be useful, for example, where a test expression specifies which portions of the documents to compare. Where, however, the test expression specifies specific values for the live document, a comparison of the documents is not required.